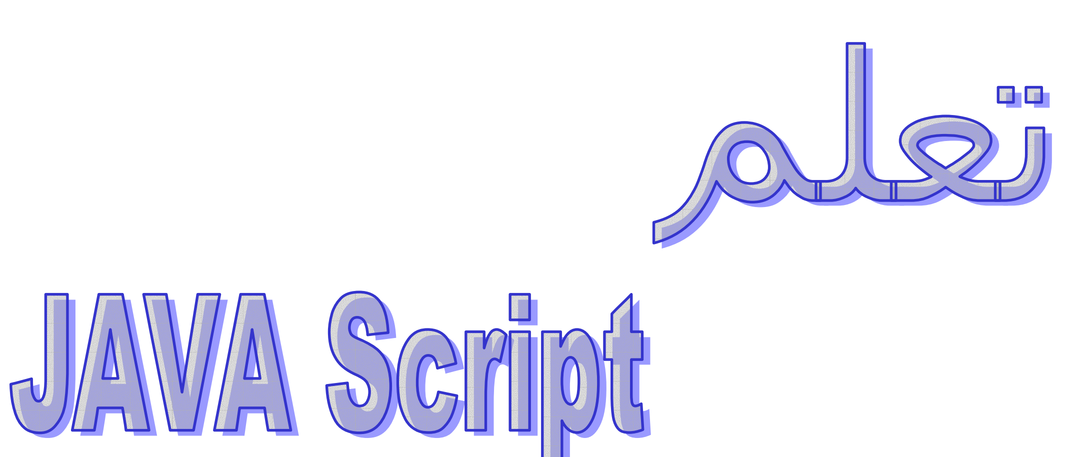


تأليف : حمود العنزي

hma2003h911@yahoo.com

الدرس الاول

بسم الله الرحمن الرحيم

نشأة الجافا سكربت :-

ظهرت في العام ١٩٩٥ م حيث كان يطلق عليها في البدايه لايف سكربت Live Script

المتطلبات لكتابه الكود المفكره او اي محرر نصوص أخر

الكتله النصيه للجافا :-

تكون على الشكل التالي

```
<SCRIPT>
<!--


//-->
</SCRIPT>
```

مع الملاحظه ان في بدايه الكتله تبدا هكذا <SCRIPT> وفي نهايتها لابد من إضافه / لها </SCRIPT>

ويتم ادخل الكود الفعلي بين السطرين الذان يبدائان بـ <!-- و //-->

استدعاء داله التحذير :-

باختصار هي الداله التي عن دخولك للموقع تظهر لك مثل هل ترشحني لافضل موقع

او الترحيب بكم في زياتكم لموقعه وهكذا

ومثالها :-

```
<SCRIPT>
<!--
alert("اهلا وسهلا بكم في منتديات الحزم الاسلاميه")؛
//-->
</SCRIPT>
```

وعند كتابتك لهذا النص عند زيارتك للموقع سوف تظهر لك هذا الرساله الترحيبيه

لاحظ علامات التنصيص لابد من وجودها في بدايه النص وفي نهايته

ولاحظ وجود الفاصله المنقوطه في اخر النص ويجب استخدامها في نهايه اي جمله

ملاحظه :-

في لغه جافا سكربت java تختلف عن JAVA مثال لو استخدمت في المثال

السابق alert وكتبتها انت حروف كبيره لن يستجيب لك داله alert

وعند اضافه // او /* و */ وهذه الرموز تفيدك لكتابة التعليقات

```
<SCRIPT>
<!--
/*hhhhhhhhhhhhhhh
hhhhhh
hhhhhhh*/
alert("7azm")؛
//hhhhhhhhhhhh
//-->
</SCRIPT>
```

لاحظ التعليقات لابد ان تبدا بـ /* وتنتهي */ او تضع // فقط

الكلمات الرئيسيه :-

وهي جزء لا يتجزء من لغه جافا سكربت

break - do - function - null - typeof - case - else - if - return - var - continue - export

import - switch - void - default - false - in - this - while - delete - for - new - true - with

المتغيرات :-

وتعتبر بمثابه المكان الذي فيه تخزن البيانات من اجل استخدامه

في اي برنامج يتم انشاؤه باستخدام جافا سكربت

عند تسميتك للمتغير لابد ان يبدا بحرف لاتيني وبامكانك تستخدم الشرطه التحتيه في اسم المتغير

اسم المتغير لا يبدا برقم

مثال لاسماء متغيرات صحيحه

JaVa

Ja_VA

java123

مثال لاسماء متغيرات خاطئه :-

hhhh٤

m88٨٨

قم بانشاء متغير جديد باستخدام الكلمه الرئيسيه "var"

```
<SCRIPT>
<!--
var message="first java"؛
alert("message")؛
//-->
</SCRIPT>
```

انواع البيانات :-

تعتبر لغه جافا سكربت بمثابه لغه صعبه في عمليه الكتابه

فان متغيراتها تقوم تخزين الارقام والسلاسل الحرفيه سواء اكانت true او false

```
<SCRIPT>
<!--
```

var a=0.06

؛"var b = "java script in easy steps

؛var c false

؛(alert(typeof a + "\n" + typeof b + "\n" +typeof c

<--//

<SCRIPT/>

المثال السابق يقوم بانشاء متغيرات ذات قيمه اوليه ثم يتم ارسالها الى داله ()alert

ويتم استخدام typeof لعرض نتيجه بنوع البيانات الخاصه بها

ويقوم الكود "+n/+" يقوم بعرض المخرجات الخاصه بكل قيمه في سطر جديد في مربع التحذير

مفهوم Escape Sequences :-

عندما تسبق علامه الشرطه المائله "/" اي حرف في نص فان

الحرف الذي يتبع هذه العلامه من نوع معين ومسئول عن احداث تاثير

محدد ولا يتعبر هذا الحرف جزء من تركيب جافا سكربت

b/-----------حذف

f/----------ادخال حرف

n/-----------سطر جديد

r/-----------علامه رجوع

t/----------علامه جدوله

'/---------علامه اقتباس مفرده ولا تدل على انتهاء السلسله الحرفيه

"/---------علامه اقتباس مزوجه تشير الى نهايه السلسله الحرفيه

//--------شرطه خلفيه واحده

ويمكن الاستفاده منهما عن طريق دمجهما داخل سلسله النص دون انهاء النص نفسه

مثال

```
alert("aaaaa\"a9aaa\"");
```

الدوال :-

وهي من اساسيات برمجه لغه الجافا سكربت

```
function call_alert(){
alert("my first java script function");
}
```

يجب ان يكون الاسم متبوعا بقوسين () ثم بزوج من الاقواس {} والتي تحتوي على الكود المراد تنفيذه

في المثال السابق اسم الداله call_alert

والكود سوف يتم استدعاء الداله alert

ويمكن لسمه "onload" الخاصه بعلامه الترميز <BODY> في HTML

```
<BODY ONLOAD="call_alert()">
```

وسائط الدوال :-

str يستخدم كاسم متغير يشتمل على بيانات سلاسل حرفيه

```
function call_alert(str){
alert(str);
}
```

```
<BODY ONLOAD="call_alert('passed value')">
```

الدوال المتعددة :-

```
function call_alert(num){
var new_number=make_double(num);
alert("the value is"+new_number);
```

{

}(function make_double(num

؛var double_num=num+num

؛return double_num

{

<"(BODY ONLOAD="call_alert(4>

نطاق المتغيرات الخاصه :-

var stored_num

}(function call_alert(num

؛stored_num=num

؛()make_triple

؛(alert("the value is"+stored_num

{

}()function make_triple

؛stored_num=stored_num+stored_num+stored_num

{

<"(BODY ONLOAD="call_alert(5>

الوسائط المتعدده :-

؛var a,b,c

}(function call_alert(str1.str2,str3

a=str1؛

b=str2:؛

c=str3؛

alert(a+b+c)؛

}

يقوم هذا المثال بتعريف الداله التي تحتوي على ثلاثه وسائط

ولذلك فان اي استدعاء لها يجب ان يحتوي على ثلاث قيم

يقوم بنقل ثلاث مجموعات من الاحرف الى الداله

*   *   *   *   *

انتهينا من الدرس الاول ولله الحمد

والان لازم تكون عرفت الاتي :-

***** طريقه كتابه الكتله النصيه**

***** استدعاء داله التحذير**

*****الكلمات الاساسيه**

***** المتغيرات**

***** انواع البيانات**

***** مفهوم Escape Sequences**

***** الدوال**

***** وسائط الدوال**

***** الدوال المتعدده**

***** المجال الخاص بالمتغير**

***** تعدد الوسائط**

الدرس الثاني وان شاء الله اعجبكم الدرس الاول :)

معاملات التشغيل الرياضيه :-

+ ====> الجمع ربط السلاسل الحرفيه ببعضها البعض

- ===> طرح

*** ===> الضرب**

/ ===> القسمه

% ===> باقي القسمه

++ ===> زياده او رفع القيمه

-- ===> انقاص او خفض القيمه

++ و -- يقومان برفع او خفض القيمه بالمقدار الذي تم تحديده

```
a=b*c-d%e/f;     \\ this is unclear
a(b*c)-((d%e)/f);   \\this is clear
var addnum=20+30؛
var addstr ="love"+"7azm"؛
var sub=35.75-28.25؛
var mul=8*50؛
var mod=65%2؛
var inc=5;inc=++inc؛
var dec=5;dec=--dec؛
var result="Addunm is"+addnum+"\n"؛
result +="Addstr is"+addstr+"\n"؛
result +="Sub is"+sub+"\n"؛
result +="Mod is"+mod+"\n"؛
result +="Mul is"+mul+"\n"؛
result +="Inc is"+inc+"\n"؛
```

result +="Dec is"+dec+"\n"؛

alert(result)؛

معاملات التشغيل المنطقيه :-

&& <=== Logical AND

|| <=== Logical OR

! <=== Logical NOT

تستخدم معاملات التشغيل المنطقيه مع المعاملات التي تحتوي على قيم حقيقه true او false

والقيم التي يمكنها ان تتحول الى true او false

& تقوم بالتحقق من صحه قيمتين ويقوم بعرض القيمه true اذا ثبت صحه كلتا القيمتين

والا فان معامل التشغيل & يقوم بعرض قيمه false

|| يقوم بعرض true اذا ثبت صحه اي قيمه من القيمتين

! يقوم بعرض ناتج اختبار او فحص القيمه المحدده لصحه المعامل او خطاه

مثال :-

لو كان المتغير JaVa يحتوي على قيمه true فان المتغير JaVa! يحتوي على قيمه false

ولافضل التطبيقات ان تكتبها بالشكل التالي JaVa=! وهذا يؤكد انه في كل خطوه يتم تغيير القيمه

```
var a=true,b=false
var test1=(a&&a); //test both operands for true
var test2=(a&&b)؛
var test3=(b&&b)؛
var test4=(a||b); //test either operand for true
var test5=(a||b)؛
var test6=(b||b)؛
```

var test7=!a;var test8=!b; //invert values

"var result="AND\n

؛result +="1:"+test1+"2:"+test2+"3:"+test3

؛"result +="\n\nOR\n

؛result +="4:"+test4+"5:"+test5+"6:"+test6

؛result +="\n\n!\n7:"+test7+"8:"+test8

؛(alert(result

معامل التشغيل الخاص بتعيين القيم :-

معامل التشغيل (=)

مثال a=b

القيمه المساويه a=b

معامل التشغيل (=+)

مثال a+=b

القيمه المساويه a=a+b

معامل التشغيل (=-)

مثال a-=b

القيمه المساويه a=a-b

معامل التشغيل (=*)

مثال a*=b

القيمه المساويه a=a*b

معامل التشغيل (=/)

مثال a/=b

القيمه المساويه a=a/b

معامل التشغيل (=%)

مثال a%=b

القيمه المساويه a=a%b

مثال :=-

var a="JavaScript", b="code"; //assign string values

a+=b;   //concatenate strings and assign to

a var c=8,d=4; //assign integer values

c+=d; //add numbers and assign result to

c var e=7.5,f=2.25; //assign float values

e-=f; //subtract f from e and assign result to

e var g=8,h=4;   //assign integer values

g*=h; //multiply numbers and assign result to

g var i=8,j=4; //assign integer values

i/=j;   //divide i by j and assign result to

i var k=8,I=4; //assign integer values

k%=I;   //divide K by I and assign result to

معاملات التشغيل الخاصه بالمقارنه :=-

== معناه التساوي

!= معناه عدم التساوي

> معناه اكبر من

< معناه اقل من

>= معناه اكبر من او يساوي

<= معناه اقل من او يساوي

مثال :=-

var teststrings1=("JavaScript"=="JavaScript")؛

var teststrings2=("JavaScript"=="javaScript")؛

var teststrings1=(1.78==1.78)؛

var teststrings2=(1.78!=1.78)؛

var teststrings1=(15<15)؛

var teststrings2=(15>15)؛

ملاحظه :=-

عند المقارنه بين الكلمات الكبتل يختلف عن سمول مثال

var teststrings1=("JavaScript"=="JavaScript")؛

هذه عباره صحيحه انهما متطابقين

var teststrings2=("JavaScript"=="javaScript")؛

هذه عباره خاطئه والسبب في اختلاف حرف جي (J) في الاولى اتى كبير وفي الثانيه صغير

معامل التشغيل الشرطي :=-

يعتبر هو معامل المقارنه المفضل

ويقوم باختبار شرطي المعادله لاثبات صحتها او خطأها وعرض نتيجه true او false

ثم يقوم بتنفيذ عباره او اثنين طبقا لنتيجه الاختبار

والتركيب الخاص لهذه هو :=-

(test expression)?if true do this: if false do this؛

يمكنك هذا المعامل من تشغيل دوال اللغه الخاصه بمتصفح ويب معين والذي يسير وفق برنامج تحديد

المتصفح

var browser=browserID;//Internet Explore or Netscape

(browser=="IE")?greetIEuser():greetNNuser()؛

****

الى هنا ينتهي الدرس الثاني وان شاء الله نلقاكم في درس جديد انتظرونا

الدرس الثالث ارجوا انكم تستفيدوا منه

عبارة if الشرطيه :=-

تركيبتها :-

```
if(test expression)statement to execulte when true؛
```

مثال /

```
var msg,num
num=7؛
if(num%2!=0){
msg="This is an odd number.";
alert(msg);
}
```

المثال السابق يتم اختبار معامل باقي القسمه (%) لتحديد اذا كانت القيمه الموجوده في المتغير num

تقبل القسمه على ٢ ام لا

***** ولاستخدام التركيب السابق للتاكد من ان الرقم فردي استخدم**

```
(num%2==1)
```

مثال يوضح الامر الخاص بـ if

```
var username=null؛
if(username==null){
username=prompt("please Enter Your Name"،"")؛
if(username!="")alert("welcom"+username)؛
```

{

عندما تنفذ المثال السابق سوف يظهر لك مربع يطلب منك دخول اسمك ثم بعد ذلك يكتب لك عباره ترحيب

بالاسم الذي انت ادخلته

العبارات البرمجيه if-else :=-

ويعرف ايضا بالتقسيم الشرطي

if(test expression)do this;else do this؛

لاحظ لا بد من وجود فاصله منقوطه قبل كلمه else

```
var num=2,bool=false؛
if(num==1&&bool==1)alert("TEST1 bool:"+bool)؛
else
if(num==2&&bool==1)alert("TEST2 bool:"+bool)؛
else
if(num==2&&bool==0)alert("TEST3 bool:"+bool)؛
else
if(num==3&&bool==0)alert("TEST4 bool:"+bool)؛
```

العبارة البرمجية switch :=-

تقوم بالتحقق من صحه الشرط ثم تقدم عنوانا يتلائم مع القيمه الناتجه

وتستخدم الكلمه الرئيسيه case في تحديد العنوان

وتستخدم الكلمه الرئيسيه default في تحديد الكود الافتراضي

ويجب ان تكون نهايه كود العنوان بالكلمه break

```
var num=2؛
```

}(switch(num

؛case 1:alert("This is case 1 code");break

؛case 2:alert("This is case 2 code");break

؛case 3:alert("This is case 3 code");break

؛("default:alert("This is default code

{

نظام التكرار for :=-

تركيبته

؛for(initializer,test,increment)statement

يستخدم initializer في تحديد قيمه البدايه للعداد الخاص بحساب عدد مرات التكرار

في الدورة وقد يستخدم المتغير لهذا الغرض كما انه يطلق عليه "i"

عندما تكون الدورة صحيحه يستمر وعندما تكون خاطئه تتوقف الدورة

كل دورة يزيد العداد

؛var a=0,b=0

}(++for(var i=0; i<5; i

؛a+=10;b+=5

{

؛(alert("FOR LOOP\n\n A is"+a+"...B is"+b

نظام التكرار For-In :=-

؛""=var i=0;a

}(or(property in window

؛"..."+a+=property

{

alert(a)؛

نظام التكرار while :=-

يتم الاختبار الى ان يصل ان عرض القيمه خطا

تحذير /*/*/

يقوم نظام التكرار اللانهائي باغلاق النص وتدمير المتصفح

var i=0,num=50؛

while(i<15){

num--؛

i++؛

}

alert("LOOP stopped at" +i+ "\nnum is now" +num)؛

نظام التكرار Do-While :=-

تستخدم الكلمه الرئيسيه do تدل على نظام التكرار

وتستمر دورة التكرار حتى تصبح النتيجه خاطئه

var i=0,num=50؛

do{

num--؛

i++؛

}while(i<1)؛

alert("LOOP stopped at" +i+ "\nnum is now" +num)؛

العبارة البرمجيه Break :=-

var i=0؛

while(i<6){

if(i==3) break؛

؛++i

{

؛(alert("BREAK\n\nLoop stopped at" +i

العبارة البرمجيه CONTINUE :=-

تستخدم في انهاء الدورة الحاليه

؛""=var i=0,str

}(while(i<5

؛++i

؛if(i==3)continue

؛""+str +=i

{

؛(alert("CONTINUE\n\nLoop stopped:" +i+ "\n\nSeries:"+str

استخدام البرمجيه With :=-

تستخدم للاشاره الى خصائص الكائن دون الحاجه الى اضافه اسم الكائن قبل اسم الخاصيه

مثال :=-

تحتوي صفحات الويب على الكائن document الذي يحتوي على الخاصيه forms للاشاره الى النماذج

الموجوده في صفحه الويب

"document.forms.order.user.value="7azm

"document.forms.order.city.value="ksa

مثال :=-

}(with(document.forms.order

؛"user.value= "7azm

```
city.value= "ksa"؛
}
```

****

اتمنى ان ينال الدرس على اعجابكم

واكون قد وفقه في توصيله اليكم

تقبلوا تحياتي

السلام عليكم ورحمه الله وبركاته

اسف جدا على التاخير في الدرس بسبب الدراسه وفقنا الله واياكم لما يحبه ويرضى

وبقي علينا عشره دروس ونكمل دروس الجافا التي عددها ١٤ درسا

نبدا بالدرس الرابع اليوم :=-

بسم الله الرحمن الرحيم

****

انشاء المصفوفات :=-

يتم التعامل مع المتغير على انه مصفوفه تستخدم اولا الكلمه الرئيسيه new

في لغه جافا سكربت مع المنشىء Array()

ويمكن تحديد انواع متعدده من البيانات في المصفوفه بتحديد

اسم المصفوفه ثم رقم الفهرس الذي يبدا بصفر

```
var a=new Array()
a[0]="First"؛
a[1]="JavaScript"؛
a[2]="Array"؛
```

الان نلاحظ ان الفهرس بدا من العنصر الصفر ورقم واحد يعتبر

العنصر الثاني ورقم اثنين يعتبر العنصر الثالث

;("var a=new Array("21st","22nd","23rd

;("،var b=new Array("Jan,","Feb,","Mar

;("var c=new Array("2001","2002","2003

;([alert(a[0]+b[1]+c[3

سوف تظهر لك داله التحذير ومكتوب فيها بالشكل هذا

st Feb 2003٢١

مع الانتباه ان رقم واحد عشرين قبل ST نفس المثال السابق تكتب

********* لتنبيه فقط *********

الان عرفت كيف تمت الطريقه اما فكرت لماذا خرج هذا الحل ام لا

الجواب بسيط جدا انظر الى الحرف a ياخذ رقم صفر اذن ناخذ العنصر الاول منه وهو st٢١

الحرف b ياخذ رقم واحد اذن ناخذ العنصر الثاني منه وهو Feb

الحرف c ياخذ رقم اثنين اذن ناخذ العنصر الثالث منه وهو ٢٠٠٣

******* انتهى التنبيه *********

عناصر المصفوفه :=-

يطلق على كل قيمه في المصفوفه اسم عنصر

خاصية length تستخدم لتحديد عدد العناصر التي تشتمل عليها المصفوفه

الخاصيه array.length تشير الى عدد العناصر في كائن المصفوفه

مثال ///

;()var a=new Array

;"a[0]="Much

;""a[1]="More

;"a[2]="JavaScript

```
؛var arraysize=a.length
(؛alert(a[0]+a[1]+a[2]+"\nNo.of elements is"+arraysize
```

استخدام نظام التكرار لادخال قيم الى عناصر المصفوفه :=-

```
؛"var str="FILL ELEMENTS LOOP\n\n
؛()var arr=new Array
}(++for (var i=0;i<8;i
؛"arr[i]="Number is"+i+"\n
؛[str +=arr[i
{
؛(alert(str
```

اضافه المزيد من العناصر :=-

لاضافه رقم الفهرس الفعلي استخدم الخاصيه array.length

المثال يوضح تعيين الرقم الصحيح ٤ للعنصر التالي الذي لا يحتوي على قيمه في المفوفه a

```
؛(var a=new Array(1,2,3
a[a.length]=4   //a.length is 3
؛(var b=new Array(5,6,7
}(++for(var i=0;i<b.length;i
[a[a.length]=b[i
{
؛(()alert(a.concat
```

سلاسل الاحرف الخاصه بالعناصر :=-

array.join() تقوم بتحويل جميع عناصر المصفوفه الى مجموعه من السلاسل الحرفيه

ويمكن ان تقبل الوسيط الاختياري لتحديد فاصل بديل للفاصله الافتراضيه

؛()var a=new Array

؛"a[0]="It

؛"a[1]="isn\'t

؛"a[2]="rocket

؛"-a[3]="science

؛"a[4]="it\'s

؛"a[5]="just

؛"a[6]="Java

؛((" ")alert(a.join

عكس الترتيب الخاص بالعنصر :=-

لو كان هذا جزء من الكتله النصيه

؛("var imgs=new Array("img1.gif","img2.gif","img3.gif

الناتج الصحيح هنا هو img3.gif-img2.gif-img1.gif

وليس img1.gif-img2.gif-img3.gif

مثال //

؛("var imgs=new Array("img1.gif","img2.gif","img3.gif

؛var rev=false

....slideshow routine is displaying the final image so//

؛rev = true

}(if(rev==true

؛()imgs.reverse

rev=false;} //remember to reset the test variable

alert(imgs.join("-")؛

المصفوفات الفرعيه :=-

array.slice() تستخدم في عمل مصفوفه جديده من المصفوفه الحاليه

مثال //

var a=new Array(1,2,3,4,5,6,7,8)؛

var b=a.slcie(2,5)؛

var c=a.slice(1,-1)؛

var d=a.slice(3)؛

ترتيب عناصر المصفوفه :=-

array.sort() تستخدم بترتيب القيم الخاصه بعناصر السلاسل الحرفيه بطريقه ابجديه

مثال //

var integers=new Array(3,8,1,9,7,5,4,2,6)؛

integers.sort(); var floats=new Array(0.5,0.125,0.57,0.25)؛

floats.sort(); var strings=new Array("7azm","mohammad","ali")؛

strings.sort()؛

var str="SORT"؛

str +="\n\nIntegers array is"+integers.join"-"؛

str +="\n\nFloats array is"+floats.join"-"؛

str +="\n\nStrings array is"+strings.join"-"؛

alert(str)؛

استخدام المصفوفه في التمهيد لعمليات التحميل الخاصه بالصور واستعراضها :=-

```
<SCRIPT>
--!>
؛("var pics=new Array("pic1.gif","pic2.gif","pic3.gif
؛()var preload=new Array
}(++for(var i=0;i<pics.length;i
؛()preload[i]=new Image
[preload[i].src=pics[i
{
<--//
<SCRIPT/>
```

المثال السابق استخدم تحميل ثلاث صور حتى تكون جاهزه

عند تشغيل التاثير الخاص بالعرض المتتالي هه الصور

المصفوفه الاولى تحتوي على العنوان الخاص بملف الصورة

السلام عليكم ورحمه الله وبركاته

نبدا الدرس الخامس

****

بسم الله الرحمن الرحيم

****

تحديد الوقت والتاريخ بما يتناسب مع زمن الاستخدام :=-

يتحدد الوقت والتاريخ اللذين يتم استخدامهما في لغه جافا سكربت من البرنامج الخاص

بالساعه والتقويم الموجودين في النظام المضيف

ولكي يمكنك الحصول على المعلومات الخاصه بساعه النظام يجب

ان يقوم النص اولا بانشاء كائن date

var now=new Date()؛

alert(now)؛

سوف ارفق صوره توضح كيفيه ظهورها انظر اليها :-)

المعلومات الخاصه بالتاريخ :=-

date.getDay تقوم بعرض اليوم كقيمه مسلسله بدايه من السبت Saturday بالقيمه صفر

وحتى القيمه ٦ الجمعه friday

يعني هكذا

السبت   ياخذ القيمه صفر

الاحد   ياخذ القيمه واحد

الاثنين   ياخذ القيمه اثنين

الثلاثاء   ياخذ القيمه ثلاث

الاربعاء   ياخذ القيمه اربعه

الخميس   ياخذ القيمه خمسه

الجمعه   ياخذ القيمه سادسا

date.getMonth تقوم بعرض الشهر كقيمه مسلسله بدايه من

january تاخذ القيمه صفر

december تاخذ القيمه ١١

var days=new Array("sat","sun","mon","tue","wed","thu","fri")؛

var mons=new Array("jan","feb","mar","apr","may","jun","jul","aug","sep","oct","nov","dec")؛

var now=new Date()؛

var yy=now.getYear()؛

var mm=now.getMonth();mm=mons[mm]؛

var dd=now.getDate()؛

var dy=now.getDay(); dy=days[dy]؛

alert(dy+""+dd+""+mm+""+yy)؛

ملاحظه اي خطا بسيط في الكلته النصيه لن ينفذ المطللوب لذلك لاحظ كل قوس قمت بكتابته لاحظ الحروف

انتبه من الاخطاء الاملائيه

المعلومات الخاصه بالوقت :=-

var now=new Date()؛

var hh=now.getHours()؛

var mn=now.getMinutes()؛

var ss=now.getSeconds()؛

var ms=now.getMilliseconds()؛

var hi="Good Morning"؛

if(hh>11)hi="Good Afternoon"؛

if(hh>17)hi="Good Evening"؛

var tim=hi+"\n"؛

tim+="Hours:"+hh+"\n"؛

tim+="Minutes:"+mn+"\n"؛

tim+="Seconds:"+ss+"\n"؛

alert(tim)؛

التوقيت الحالي :=-

date.getTime() تستخدم للمقارنه بين اثنين من الكائنات الخاصه بالتاريخ من الناحيه العديده

```
var start = new Date()؛

var msec1=start.getTime()؛

var num=0؛

for(var i=0;i<250000;i++){

num++؛

}

var stop=new Date()؛

var msec2=stop.getTime()؛

var diff=(msec2-msec1)/1000؛

alert("Time elapsed:"+diff+"seconds")؛
```

****
ملاحظه // اذا كان متصفحك netscape قم بتقليل عدد مرات التكرار من ٢٥٠٠٠٠الى ٥٠٠٠
****

التعرف على Universal Time Clock :=-

يعتبر Universal Time بمثابه التوقيت العالمي توقيت خط جرينتش

```
var now=new Date()؛

var hh=now.getHours()؛

var mn=now.getMinutes()؛

if(mn<=9)mn="0"+mn

var ss=now.getSeconds()؛

var utc_hh=now.getUTCHours()؛

var wet="athens time:"+hh+":"+mn+":"+ss+"\n\n"؛
```

;var utc="Universal time:"+utc_hh+":"+mn+":"+ss

;(alert(wet+utc

يقوم المثال باستدعاء التوقيت المحلي للغرب الاوربي

****
ملاحظه / اذا كانت الدقائق اقل من عشره فان القيمه ستكون عباره عن رقم واحد

ولذالك فان نص البرمجه سيقوم باضافه صفر اليها
****

المنطقه الزمنيه :=-

يمكن للغه جافا سكربت ان تحدد المنطقه الزمنيه التي ينتمي اليها المستخدم من خلال

التقنيه date.getTimeOffest

;()var now=new Date

;()var offset=now.getTimezoneOffset

;var msg

}(switch(offset

;case 240:msg="East Coast";break

;case 300:msg="Central";break

;case 360:msg="Mountain";break

;case 420:msg="Pacific";break

;"default:msg="all

{

;(".alert("Welcome to"+msg+"visitors

ادخال قيم لتحديد التاريخ والوقت :=-

;()var now=new Date

var orig="Original:\n"+now.toString()+"\n\n"؛

now.setDate(21)؛

now.setMonth(1)؛

now.setHours(12)؛

now.setMinutes(30)؛

now.setSeconds(15)؛

now.setFullYear(2005)؛

var mod="Modified:\n"+now؛

alert(orig+mod)؛

****
toString تقوم بعرض قيمه عباره عن سلسله حرفيه
****

اسلوب تخزين القيم الخاصه بكائن date :=-

date.toGMTString تقوم بتحويل القيم الى سلاسل حرفيه وتقوم بتحويلها الى المنطقه الزمنيه GMT

var now=new Date()؛

VAR jpn="Tokyo time:\n"+now.toString()+"\n\n"؛

VAR gmt="UTC time:\n"+now.toGMTString()؛

alert(jpn+gmt)؛

استخدام ساعة النظام :=-

ساعه النظام يمكن استخدامها مع تقنيه setTimeout()

window.setTimeout() تقوم باستخدام تقنيه من اجل عرض مربع حوار

التحذير في فتره تصل الى عشر ثواني

```
؛var num=0
}()function annoy
؛++num
؛(alert("This is 10-Second Message No:"+num
؛(window.setTimeout("annoy()",10000
{
؛()annoy
```

الغاء المؤقت الزمني :=-

```
؛var num=0,tim
}()function advise
؛++num
}(if(num==3
؛(alert("Ok-Itold you twice");   window.clearTimeout(tim
}else{
؛(alert("This is 10-Second Message No:"+num
{؛(tim=window.setTimeout("advise()",10000
{
؛()advise
```

اهميه نظام جافا سكربت في clock :=-

```
<HTML>
<"()BODY ONLOAD="tick>
```

```
<SCRIPT>
--!>
}()function tick
؛()var now=new Date
؛var hh=now.getHours(); if(hh<=9)hh="0"+hh
؛var mn=now.getMinutes(); if(mn<=9)mn="0"+mn
؛var ss=now.getSeconds(); if(ss<=9)ss="0"+ss
؛var tt=hh+":"+mn+":"+ss; document.f.clock.value=tt
؛(window.setTimeout("tick()",1000
{
<--//
<SCRIPT/>
<"FORM NAME="f>
<"INPUT NAME="clock"TYPE="text"SIZE="10>
<FORM/>
<BODY/>
<HTML/>
```

انظر الى الصوره الثانيه توضيح لهذا المثال

****

والله يوفقنا واياكم لما يحبه ويرضا

ونصر اخواننا المجاهدين في كل مكان

اللهم آمين

اخوكم

حمود العنزي

HMA2003H911@YAHOO.COM

بسم الله الرحمن الرحيم

نبدأ في الدرس السادس من دروس الجافا سكربت

يحتوي كائن Math في لغة جافا سكربت على عدد من القيم

الرياضية القياسية الثابتة لتسهيل الإشارة إليه

وتستخدم الأحرف الكبيرة في الإشارة إلى هذه القيم الثابتة المدرجة في الجدول :=-

الجدول فقط من غرض المعرفه لا غير :)

ولكن الجدول سوف اقوم بارفاقه في الرساله الثانيه بملف وورد ان احببت الاطلاع عليه :-)

وتستخدم الثوابت في الكائن Math فقط في تطبيقات الخاصه بلغه جافا سكربت

استخدام الثابت PI :=-

يقوم الثابت Math.PI بانشاء امكانيات جيده وتاثيرات فعاله

```
var pos=new Array();   //array for position data

var xos=30,yos=30;  //xy coordinate offsets

var rad=40;    //radius of circle

}()function init

}(++for(var i=0;i<4;i

pos[i]+=Math.PI/45;    //increment degrees

؛(([var x=xos+(rad*Math.cos(pos[i

؛(([var y=yos+(rad*Math.sin(pos[i

؛(if(document.layers)document.layers["lyr"+i].moveTo(x,y

(if(document.all
```

```
eval("lyr"+i+".style.top=x;lyr"+i+".style.left=y")؛
}
setTimeout("spin()",100)؛
}
```

تقنيه Math.cos و Math.sin تعتبر من مجموعه تقنيات الكائن Math

التقنيات الخاصه بالكائن Math :=-

انظر الى الجدول الثاني المرفق في الرد الثاني ((مع الجدول الاول :-)))

تقريب الارقام :=-

تستخدم تقنيه Math.round في تقريب الاعداد التي تشمل على علامه عشريه

بحيث تصل الى اقرب رقم صحيح وايضا ابدال الاعداد العشريه الطويله

الى اعداد اقصر

مثال :=-

```
var a=7.5؛
a=Math.round(a)؛
a="Rounded positive:"+a+"\n"؛
var b=-7.5؛
b=Math.round(b)؛
b="Rounded Negative:"+b+"\n"؛
var c=3.764638467915؛
c=c*100    //take it up two places
c=Math.round(c);    //do the round
```

c/=100;   //take it back down 2 places

؛c="Commuted Long Float:"+c

؛(alert(a+b+c

****
ملاحظه // الرقم السالب لا يقربه الى الذي اكبر منه بل بالذي اقل منه

في المثال السابق اراد تقريب -٧,٥ تقريبها هو -٧
****

تقريب الارقام السالبه والموجبه :=-

تقنيه Math.ceil وهي تقنيه تساعد على تقريب الرقم العشري الى اكبر رقم صحيح

تقنيه Math.floor وهي تقنيه تساعد على تقريب الرقم العشري الى اصغر رقم صحيح

؛var a=7.5

؛(var a1="a1:"+Math.ceil(a

؛(var a2="a2:"+Math.floor(a

؛var b=-7.5

؛(b=Math.round(b

؛(var b1="b1:"+Math.ceil(b

؛(var b2="b2:"+Math.floor(b

؛"var rup="ROUNDING UP\n"+a1+"\n"+b1+"\n\n

؛var rdn="ROUNDING DOWN\n"+a2+"\n"+b2

؛(alert(rup+rdn

****
عند تقريب القيم السالبه الى اقرب رقم صحيح فان التقريب يصل الى الصفر
****

مقارنه الارقام :=-

التقنيه Math.max تقوم بمقارنه قيمتين كوسيطين ثم تقوم بعرض القيمه الاكبر

في الاثنين الفرق بين القيمتين

التقنيه Math.min نفس السابقه ولكنها تقوم بعرض اقل القيمتين

Math.pow تستخدم لانشاء قيم مربعه ومكعبه للمقارنه

```
var sq=Math.pow(5,2)؛
var cb=Math.pow(3,3)؛
var hi="Round up:"+Math.max(sq,cb)؛
var lo="Round Down:"+Math.min(sq,cb)؛
var ng="Round Negative Up:"+Math.max(-5,-4.75)؛
alert("MAXMIN\n\n"+hi+"\n"+lo+"\n"+ng)؛
```

حصر القيمه بين عددين عشوائيين :=-

تقنيه Math.random تستخدم لعرض قيمه عباره عن رقم عشري محصور بين ٠,٠ و ٠,١

فاذا كان المضروب ٤

سيعمل على انشاء رقم عشوائي يتراوح بين ٠,٠ و ٤,٠

واذا كنت ترغب في التقريب استخدم Math.ceil

```
var rand1=Math.random()؛
var rand2=rand1*10؛
var rand3=math.ceil(rand3)؛
var another=Math.ceil(Math.random()*10)؛
```

الارقام العشوائيه :=-

```
var n=new Array(6)؛

for(var i=0;i<6;i++){n[i]=""؛}

while(n[5]==""){

var j=Math.ceil(Math.random()*49)؛

if(n[0]=="")n[0]=j؛

else

if(n[1]==""&&j!=n[0])n[1]=j؛

else

if(n[2]==""&&j!=n[0]&&j!=n[1])n[2]=j؛

else

if(n[3]==""&&j!=n[0]&&j!=n[1]&&j!=n[2])n[3]=j؛

else

if(n[4]==""&&j!=n[0]&&j!=n[1]&&j!=n[2]&&j!=n[3])n[4]=j؛

else

if(n[5]==""&&j!=n[0]&&j!=n[1]&&j!=n[2]&&j!=n[3]&&j!=n[4])n[5]=j؛

}

var str=""

for(var i=0;i<6;i++){str+=n[i]+""؛}

alert("LOCKY NUMBERS\n\n"+str)؛
```

***
عند تنفيذك لهذا المثال عند كل تحديث للصفحه يطلع لك رقم مختلف
***

الصور العشوائيه :=-

يتم ذلك من خلال علامه الترميز img لخاصه بـ Html وتحديد قيمه pic في سمه scr

```
var pix=new Array("pic0.gif","pic1.gif","pic2.gif","pic3.gif","pic4.gif","pic51.gif","pic6.gif")؛

var rand=Math.floor(Math.random()*7)؛

document.images.pic.scr=pix[rand]؛

document.write(rand+":"+pix[rand])؛
```

****

انتهينا من الدرس السادس وانتظروا الدروس المتبقيه

اخوكم :=- حمود JaVa-ScRipT

hma2003h911@yahoo.com

جدول Math

الثابت	ما يدل عليه
Math.E	الثابت e أساس اللوغاريتم الطبيعي الذي تبلغ قيمته تقريبا ٢,٧١٨٢٨
Math.LN2	اللوغاريتم الطبيعي لعدد ٢
Math.LN10	اللوغاريتم الطبيعي لعدد ١٠
Math.LOG2E	اللوغاريتم الأساس ٢ للثابت e
Math.LOG10E	اللوغاريتم الأساس 10 للثابت e
Math.PI	الثابت pi
Math.SQRT1_2	القيمة المعكوسة للجذر التربيعي للأساس ٢

Math.SQRT2	الجذر التربيعي للأساس ٢

التقنيات الخاصة بالكائن Math :- الجدول الثاني ☺

Math.abs()	عرض القيمة المطلقة
Math.acos()	يكون الناتج عبارة عن جيب تمام الزاوية
Math.asin()	يكون الناتج عبارة عن جيب الزاوية
Math.atan()	يكون الناتج عبارة عن ظل الزاوية
Math.atan2()	نقل الزاوية من المحور x إلى نقطة
Math.ceil()	التقريب لأقرب عدد موجب
Math.cos()	يقوم بعرض قيمه جيب تمام الزاوية
Math.exp()	الناتج عبارة عن أس ثابت e
Math.floor()	التقريب لأقرب عدد سالب
Math.log()	اللوغاريتم الطبيعي
Math.max()	الناتج عبارة عن أكبر العدديين
Math.min()	الناتج عبارة عن أصغر العدديين
Math.pow()	تقوم بعرض قيمه المقدرة
Math.random()	الناتج عبارة عن رقم عشوائي
Math.round()	التقريب لأقرب عدد صحيح
Math.sin()	تقوم بعرض قيمه جيب الزاوية
Math.sqrt()	الناتج عبارة عن الجذر التربيعي
Math.tan()	الناتج عبارة عن ظل الزاوية

السلام عليكم ورحمه الله وبركاته

اعتذر لتاخري في الدرس وذلك لاسباب الاختبارات

نبدا في الدرس السابع

بسم الله الرحمن الرحيم :=-

****

أشكال السلاسل الحرفيه :=-

عاده ما تتميز بوجود علامات الاقتباس المزدوجه

var a= "JavaScript Strings"؛

alert(a.length)؛

يقوم المثال بالمقارنه بين خاصيه array.length

وخاصيه string.length

****

يتم التعامل مع المسافات على انها احرف كبيره

****

مكونات السلاسل الحرفيه :=-

تعتبر بمثابه مصفوفات يكون كل عنصر بمثابه حرف يمكن الاشاره

اليه بنفس الطريقه التي يشار بها الى عناصر المصفوفه المنظمه

ويمكنك العثور على هذه الاحرف باستخدام الخاصيه string.chartAt()

التي تتعامل مع رقم فهرس العنصر كوسيط خاص بها

var str="linger in",newstr=""؛

var a="First letter:"+str.charAt(0)؛

var z="Final letter:"+str.charAt(str.length-1)؛

for(var i=0;i<str.length;i++){

؛(if(str.charAt(i)!="i")newstr+=str.charAt(i

"else newstr+="o

{

؛"var result="STRINGS\n\n

؛"result +="New string:"+newstr+"\n

؛(alert(result+a+"\n"+z

ربط السلاسل الحرفيه :=-

يستخدم المعامل "+" في الربط بين السلاس الحرفيه ويستخدم على نطاق واسع في الكود الخاص

؛var num=99

؛var flt=98.6

؛var bool=true

؛"var str1="Although

؛""\var str2="F is cited as \"body temperatrue

؛"var str3="\nthe

؛"var str4"range is from about 97 to

؛".var str5="F

؛(alert(str1+flt+str2+str3+bool+str4+num+str5

البحث عن حرف معين داخل سلسله حرفيه :=-

استخدم الخاصيه string.index تستخدم في البحث عن اي حرف

او سلسله حرفيه فرعيه يتم تعيينها كوسيط

وفي حاله حدوث التطابق تقوم التقنيه بعرض اول موضع يرد فيه الحرف المطابق او السلسله الحرفيه

المطابق او السلسله الحرفيه الفرعيه داخل السلسله الحرفيه التي يتم البحث عنها

واذا لم يحدث اي نوع من التطابق فان التقنيه السابق تقوم بعرض القيمه -١

وتستخدم الخاصيه string.indexOf للبحث عن النقط وكذلك الحرف @ داخل السلسله الحرفيه

الخاصه بالبريد الاليكتروني بهدف التاكد من صحتها

```
var str="mailme@domain.com"؛
var att=str.indexOf("@")؛
var dot=str.indexOf(".")؛
var hsh=str.indexOf("#")؛
var fmt=((att!=-1)&&(dot!=-1))?"valid":"invalid"؛
var res="Format is" +fmt؛
res +="\n@ at" +att
res +="\nDot at" +dot
res +="\n\nHash is" +hsh
alert(res)؛
```

تقسيم السلسله الحرفيه الواحده :=-

يتم فصل مكونات السلسله الحرفيه باستخدام تقنيه string.split

```
var str="JavaScript in easy steps"؛
var ss=str.split("")'
var bk="JavaScript in easy steps|Mike McGrath|$9.99"؛
var bks=bk.split("|")؛
var res="Topic :" +ss[0]؛
res +="\nTitle :" +bks[0]؛
res +="\nAuthor :" +bks[1]؛
```

؛[res +="\nPrice :" +bks[2

؛(alert(res

السلاسل الحرفيه الفرعيه :=-

يمكنك الحصول على سلسله حرفيه فرعيه من السلسله الحرفيه الحاليه

باستخدام التقنيه string.substring

التي يجب ان تحتوي على وسيطين لتحديد مواضع البدايه والنهايه للسلسله

الحرفيه الجديده المطلوبه داخل السلسله الحرفيه الاصليه

ويمكنك عرض السلسله الحرفيه الفرعيه من السلسله الاصليه باستخدام التقنيه string.substr

؛"var str="JavaScript in easy steps

(var sub1=str.substring(14,19)+str.substring(0.11

؛(var sub2=str.substr(14,5)+str.substr(0.10

؛var res="Substring method :" +sub1

؛res +="\nSubstr method :" +sub2

؛(alert(res

التغيير الى السلاسل الحرفيه :=-

object.toString() تقوم بتحويل انواع اخرى من البيانات الى سلاسل حرفيه من خلالها

وتعتبر هذه الداله اساسيه والهدف منها دائما عرض السلسله الحرفيه

الخاصه بالكائن الذي تم الاستعلام عنه حتى لو لم يكن هذا الكائن نفسه بمثابه سلسله حرفيه

كما يمكنك استخدام هذه التقنيه في اي نص مكتوب بلغه جافا سكربت

من اجل الكشف عن بعض المعلومات التي تتعلق بالكائن

var num=9؛

var nostr=num+9؛

var adstr=num.toString()+9؛

var img=new Image()؛

var res="No strings :"+nostr؛

res +="\nAdded strings :"+adstr؛

res +="\nImage :" +img.toString()؛

alert(res)؛

تغيير حاله الاحرف :=-

يمكنك تغيير حاله الاحرف في السلسله الحرفيه باستخدام احدى هذه التقنيتين

string.toUpperCase() او string.toLowerCase

var entry=document.forms[0].textfield.value؛

if(entry.toLowerCase() =="castle")alert("Accepted")؛

else alert("Refused")؛

الحصول على الاعداد من السلاسل الحرفيه :=-

يمكنك استخراج عدد من بدايه السلسله باستخدام تقنيه

parseInt() و parseFloat()

parseInt() تقوم بتحليل السلسله الحرفيه وعرض اول قيمه صحيحه تم تعيينها

parseFloat() تقوم بنفس العمل وبالطريقه نفسها ولكنها تقوم

بعرض اعداد تشتمل على علامه عشريه

isNaN() تقوم بالبحث عن قيمه غير عدديه

var str="66.6% PASS RATE"؛

```
var res="Integer :" +parseInt(str);
var res="\nFloat :" +parseFloat(str);
var badstr="PASS RATE66.6%";
res +="\nNon-numeric :"+parseInt(badstr);
res +="\nFound :";
res +=isNaN(parseInt(badstr))?"Character":"Number";
alert(res);
```

عرض قيم المتغيرات في السلاسل الحرفيه :=-

eval() تقوم بتعيين وسيط عباره عن سلسله حرفيه تحتوي على الكود الذي تمت كتابته بلغه جافا سكربت

والمراد التحقق منه التقرير بصحته او خطاه

```
var inputname="textfield";
var inputtext="\"JavaScript Text\"";
eval("document.forms[0]."+inputname+".value="+inputtext);
```

كتابة اكواد للسلاسل الحرفيه :=-

string.ChartCodeAt() تتطلب هذه التقنيه تعيين فهرس لكل حرف داخل السلسله كوسيط لها

string.fromChar Code() هذه التقنيه عكس السابقه تمكنك من الحصول على الاحرف من الكود الموحد

```
var str="Code Fun";
var enc="";
var unc="";
for(var i=0;i< str.length;i++){
```

```
؛"،"+(enc += str.charCodeAt(i

{

؛("،")var ss=enc.split

}(++for(var i=0;i< ss.length;i

؛([unc +=String.fromCharCode(ss[i

{

؛(alert("Encoded:"+enc+"\nUnencoded:"+unc
```

****
****
****
****
****

والى هنا نصل الى نهايه الدرس السابع

وارجوا ان اكون قد افدتكم وانتظروا باقي الدروس

واعذرونا على التاخير بسبب اختبارات الجامعه

تحياتي اخوكم

حمود العنزي

hma2003h911@yahoo.com

السلام عليكم ورحمه الله وبركاته

كل عام وانتم بخير

لا تنسوا اخواننا واهلنا في فلسطين بالدعاء في هذا الشهر الفضيل

وبقي علينا ستة دروس ونكمل دروس الجافا التي عددها ١٤ درسا

نبدا بالدرس الثامن اليوم :=-

بسم الله الرحمن الرحيم

****

منشئ الكائن :=-

ويمكن انشاء كائنات مخصصة جديدة بسهولة من خلال الكلمة الرئيسية new

في لغه جافا سكربت والمنشئ Object()

المثال التالي يقوم بأنشاء كائن جديد باسم car ويقوم بتحديد ثلاث خصائص لهذا الكائن ذات قيم محددة

var car=new Objcet()؛

car.maker="Porsche"؛

car.model="Boxster"؛

car.color="Red"؛

alert(car.color+""+car.maker+""+car.model)؛

****
لاحظ وجود مسافه بين نيو وبروجكت new Objcet
****

مثال اخر يقوم بتناول كيفية تحديد كائن جديد لخاصيه في

كائن ثان مخصص وكيف يمكن عرض الخصائص المكتسبه

var mondeo=new Object()؛

mondeo.badge="Mondeo"؛

mondeo.body="Saloon"؛

mondeo.doors=4؛

mondeo.color="Red"؛

mondeo.engine="1.8litre"؛

var car=new Objcet()؛

car.maker="Ford"؛

car.model=mondeo؛

alert(car.maker+""+car.model.badge+""+car.model.body)؛

التسلسل الخاص بــ DOM

يشير Document Object Model(DOM) في اي متصفح مجموعة من الكائنات في

هذا المتصفح يمكن عرضها من خلال جافا سكربت من اجل التاثير على

اداء المستند الذي تم انشاؤه بلغه HTML

وتمتلك DOM خصائص حتى يمكنها ان تشتمل على مصفوفة من العناصر في الويب مثل images[] مع

document.images[] كخاصيه في كائن document الذي سيقوم

بتخزين عنوان كل صورة موجودة في صفحة الويب

ويتم تخزين العنوان الخاص بالصورة الاولى في الكود الذي تمت كتابته بلغة html داخل

المصفوفه document.images[0] ثم يتم تخزين العنوان الخاص بالصور المتتابعه في فهارس

العناصر الخاصه بالمصفوفه المتزايده

ولكن لا يعتبر DOM بمثابه مقياس بين ادوات التصفح

لذلك Netscape DOM تختلف عن DOM

مثال

يقوم هذا المثال بتعيين قيمه للخاصيه document.bgColor

document.bgColor = "black";

****
عند تنفيذك للكود السابق سوف ترى ان الخلفيه اصبح سوداء
****

****
يمكنك استخدام تركيب self لعرض window مثل self.bgColor
****

استعراض كائنات الصفحة :=-

يوضح النص البرمجي التالي كائنات window الافتراضيه ي كل من اصداري المتصفحين

Netscape 4.7 و Internet Explorer 5.5

(for(propertyName in window

؛("،"+document.write(propertyName

****
غير كلمه window في الكود في السطر الاول بكلمه document لمعرفه المزيد عن كائنات DOM
****

نوع المتصفح :=-

يحتوي كائن navigator على خصائص توفر المعلومات المتعلقه بالمتصفح الذي يتم استخدامه

لعرض المستند

وتقوم خاصية navigator.appName بعرض اسم المتصفح

وتقوم خاصية navigator.appVersion بعرض رقم الاصدار الخاص بالمتصفح بالإضافه تحتوي على مزيد من

التفاصيل الاختياريه

مثال =:- يقوم المثال القادم بعرض المعلومات الاضافيه للمستخدم خاصة بنظام التشغيل

كما يعرض المتصفح الخاص بك

؛var browser = navigator.appName

؛var version = navigator.appVersion

؛(alert(browser+"\n"+version

****
في بعض الاحيان يخرج لك نوع المتصفح والاصدار فقط مع اضافه اخر الجمله بحرف "i" وهذا يدل على مستوى

التشفير المتبع عالميا وهو خاص بالنت سكاب Netscape
****

التعرف على المتصفح :=-

اضافه خصائص جديده الى الكائن document لتشغيل المؤثرات الديناميكيه في dhtml

شركه Netscape قدمت خاصيه document.layers

بينما شركة مايكروسوفت Microsoft بانشاء خاصيه document.all

ومن الافضل ان تقوم بفحص هذه الخصائص من اجل التعرف على نوع المتصفح والتاكد من قدرته على الكتابه

بلغه dhtml

يقوم المثال التالي باختبار ادوات التصفح وفتح الصفحة الخاصه بالمتصفح طبقا لادوات التصفح الحديثه

والقديمه الخاصه بشركة Netscape وشركه Microsoft او الصفحه الافتراضيه الخاصه بايه

ادوات تصفح اخرى

```
if(document.all)window.location = "ie-dhtml.html"؛

if(document.layers)window.location = "nn-dhtml.html"؛

else {

var oldnn = (navigator.appName == "Netscape"&&
(parseInt(navigator.appVersion)<4)؛

location = (oldnn) ? "nn-old.html" : "default.html"؛ }

}
```

فحص نظام التشغيل :=-

تعتبر ادوات التحكم الخاصه بــ active x بمثابه احدى الطرق المستخدمه في

انشاء سمات الواجهة على سمات الويب

يقوم المثال القادم بكتابة سطر من النص طبقا لنظام التشغيل الذي تحدده

؛"if(navigator.platform == "Win16")var os="Windows3.1

؛"if(navigator.platform == "Mac")os="MacOS

؛"if(navigator.platform == "Win32")os="Windows95

؛(document.write("Operating system is" +os

برنامج Schockwave :=-

قد تقوم لغه جافا سكربت بعمليه اختبار او فحص لمعرفه ما اذا كان بامكان المستخدم

ان يقوم بعرض فيلم Flash Macromedia وتحميل ايه صفحة بديله اذا كان برنامج Schockwave

غير متاح

ويمكن استخدام navigator.plugin في ادوات التصفح الخاصه بالنت سكيب من

اجل الحصول على برنامجي فلاش و SchockKwave وهذه الطريقه لا يمكن تدعيمها

بشكل كامل في متصفح انترنت اكسبلور

يقوم المثال التالي بعمليه الفحص ثم يقوم بتحميل الصفحة المحددة طبقا للنتيجة

"if((navigator.appName == "Microsoft Internet Explorer

navigator.appVersion.indexOf("Mac") ==-1&&

(navigator.appVersion.indexOf("3.1") == -1&&

||

(["navigator.plugins&&navigator.plugins["Shockwave Flash)

navigator.pulgins["Shockwave Flash 2.0"]) window.location = || ؛""playflash.html

else

window.location = "noflash.html";

اسلوب التعامل مع الاخطاء في لغه جافا Java :=-

يمكن تقيم مستوى كفاءة المتصفح باستخدام لغه جافا من خلال تقنية navigator.javaEnabled()

لعرض القيمه البوولينيه صحيحة كانت ام خاطئة

وتعتبر بمثابه تقنيه خاصة بكائن navigator وليست مجرد خاصيه ولذلك فهي يجب ان تكون متبوعه باقواس

مثال

```
if(navigator.javaEnabled())
window.location = "javapage.html";
else
window.location = "nonjavapage.html";
```

****
****
****
****
****
****

انتهى الدرس الثامن وعن قريب سوف اضع الدرس التاسع لاني كتبته

بس ما كملته الا الان باقي عليه قليل جدا

اشكر جميع الاخوان الذين راسلون لطلب ارسال الدروس لهم او الاسئله او الشكر

جعلنا الله واياكم في هذا الشهر الفضيل ممن يقوم رمضان ايمانا واحتسابا

ولا تنسوا اخوانكم من فلسطين بالدعاء في هذا الشهر الفضيل

اخوكم حمود العنزي

hma2003h911@hotmail.com

السلام عليكم ورحمة الله وبركاته

هذا الدرس التاسع من دروس جافا سكربت

وارجوا ان ينال على رضاكم

بسم الله الرحمن الرحيم

---*---*---*---*---*---*---*---

عملية التاكيد :=-

تعتبر خاصيه alert() بمثابه تقنيه تابعه لكائن window

ويمكن كتابتها كالاتي

window.alert()

وتعتبر تقنيه window.confirm بمثابه التقنيه الاولى في التقنيتين الخاصتين بكائن window

والتي تعرض مربعات نصيه عند استدعائها ويمكن استخدامها للحصول على تاكيد من المستخدم

في مربع الحوار الذي يحتوي على زر موافق ok الذي يقوم بعرض true وزر cancel الذي يقوم

بعرض false للنص البرمجي

مثال

```
var ask = comfirm("Do you wish to proceed؟")؛
var msg؛
if(ask)msg="OK button was pushed"؛
else msg = "Cancel button was pushed"؛
```

document.write(msg)؛

****
في هذا المثال تعتبر ask بمثابه اختصار لـ (ask==true)
****

عمليه الادخال :=-

تستخدم تقنيه window.prompt() لاتاحه امكانيه ادخال البيانات الخاصه بالمستخدم

من لوحه المفاتيح عبر مربع حوار يحتوي على زرين OK و Cancel

وتحتاج هذه التقنيه الى اسم المستخدم ولتحديد الرساله التي سيتم عرضها

وقيمه افتراضيه خاصه بحقل النص واذا لم يتوافر اسم المستخدم في النص الافتراضي

فلن يتم تحديد الرساله

مثال

```
var user = prompt("Please enter your name...","")؛

if(user==null||user=="")user="visitor"؛

document.write("Hi"+user+",welcom to this page")؛
```

****
طبعا تستطيع تغيير السؤال الى اللغه العربي

وكذلك الناتج على حسب ما تريد انت

وكلمه visitor هذه تخرج اذا لم يكتب المستخدم اسم وتستطيع تغييرها الى كلمه زائر بالعربي

او مجهول كما هو موجود في اغلب الصفحات
****

اهميه تحديد موضع الاطار :=-

يحتوي كائن window.location على العنوان الكامل الخاص بالمستند الذي يتم تحميله في المتصفح

كما انه يقوم تخزين العنوان الكامل في خاصيه window.location.herf وتحديد قيمه جديده

له او الكائن window.location نفسه الذي سيقوم بتحميل العنوان في المتصفح

ويتكون العنوان المثالي من

protocol://host/pathname?#hash

مثال في هذا المثال ينتقل location.hash الى الموقع المحدد في الصفحه

```
var locn = "Href:"+location.href؛
var port = "Protocol:"+location.protocol؛
var host = "Host:"+location.host؛
var path = "Path:"+location.pathname؛
var hash = "Hash:"+location.hash
alert(locn+"\n"+port+"\n"+host+"\n"+path+"\n"+hash)؛
```

الانتقال بين الصفحات :=-

يحتوي كائن window.history على مصفوفه خاصه بعناوين المواقع التي قمت بزيارتها سابقا

في احدى نوبات العمل داخل المتصفح

وتقوم تقنيتي back() و forward() الخاصه بكائن window.history بنفس وظيفيه ازرار

Backward و Forward في المتصفح

ويمكنك الحصول على مزيد من المرونه في عمليه الانتقال باستخدام تقنيه

window.history.go() التي تقوم بتحديد وسيط واحد من اجل تحديد الموقع من خلال عنوان تقريبي

وفي حاله قيام المستخدم بحذف الصفحه السابقه فان هذا المثال سيقوم باسترجاع

الصفحه السابقه من اجل تكمله حقل النص

if(required_input=="")history.go(-١)؛

التحميل :=-

يستخدم كائن window.onload في تحديد اسم الداله التي يتم استدعاؤها في الحال بعد ان

يتم تحميل المستند في المتصفح

وتقوم الداله init() بتعيين قيمة للكائن وغالبا ما تستخدم في تعيين عدد من القيم داخل

نص البرمجه

وتقوم سمه onload في html التي قد تكون مدرجه في body بتحديد داله onload

وينطبق هذا على كائن window.onunload وسمه onload التي تقوم بتحديد الداله بهدف استدعائها

عندما ينتقل المستخدم الى الموقع الجديد

ويقوم كائن window.onunload باستدعاء الداله التي تقوم بانشاء

اطار عندما يغادر المستخدم موقع الويب

في المثال التالي يقوم بعرض مربع حوار التحذير عندما يخرج المستخدم من هذه الصفحه

```
function goodbye(){
alert("Thanks for calling - Come back soon")؛
}
window.onunload = goodbye؛
```

استخدام شريط الحاله :=-

يمكن استخدام كائن window.status لعرض الرساله في شريط الحاله الخاص

باطار المتصفح الى ان تقوم سمه اخرى في المستند المعروض بوضع رساله جديده

في شريط الحاله او يتم ارجاع شريط الحاله الى الوضع الافتراضي

يمكنك تحديد الرساله الافتراضيه باستخدام window.defaultStatus

مثال

يقوم هذا المثال الذي يستخدم جافا سكربت داخل علامات لترميز html بتحديد الرساله

الافتراضيه واضافه onmouseover للرابط الذي يقوم بعرض رساله اخرى اثناء وجود المؤشر

على الرابط

<"'BODY ONLOAD = "window.defaultStatus = 'Status Demo>

"A HREF ="nextpage.htm>

<"ONMOUSEOVERN ="window.status='Click Here';return true

<Hypertext Link</A

****
المثال السابق تكتبه بلغه الترميز html

يعني بالمختصر هكذا يكتب المثال السابق

<html>

<"'BODY ONLOAD = "window.defaultStatus = 'Status Demo>

"A HREF ="nextpage.htm>

<"ONMOUSEOVERN ="window.status='Click Here';return true

<Hypertext Link</A

<html/>
***

الاطارات المنبثقه :=-

فتح اطار ثان في المتصفح لعرض عنوان موقع آخر

ويمكن حدوث ذلك من خلال الاستعانه بتقنيه window.open() التي تتطلب ثلاثه

وسائط لتحديد عنوان الموقع الذي سيتم تحميله في الاطار المنبثق وكذلك اسم الاطار

والسمات المطلوبه

يقوم المثال التالي بفتح اطار فرعي والذي يتبع في تحميله نفس خطوات الاطار الاول

```
function popup(){
window.open("popup.html","",
"top=40,left=40,width=200,height=100");
}
window.onload=popup;
```

****
غير الطول والعرض ومكان العرض فوق او تحت او يمين او يسار حسب ما تريده انت
***

سمات الاطارات المنبثقه :=-

السمه Directories وظيفتها اضافه شريط الروابط

السمه Height وظيفتها تقوم بتحديد الارتفاع في مساحه المستند باستخدام وحدات البكسل

السمه Left وظيفتها احداثي x الخاص بالاطار على الشاشه

السمه Location وظيفتها اضافه شريط العنوان

السمه Menubar وظيفتها اضافه اشرطه القوائم القياسيه

السمه Resizable وظيفتها تسمح بالتحكم في حجم الاطار

السمه Scrollbars وظيفتها الحصول على اشرطه التمرير عند الحاجه اليها

السمه Status وظيفتها اضافه شريط الحاله

السمه Toolbar وظيفتها اضافه شريط الازرار مع كل من زر back و forward

السمه Top وظيفتها احداثي y الخاص بالاطار على الشاشه

السمه Width وظيفتها تقوم بتحديد العرض في مساحه المستند باستخدام وحدات البكسل

تحتاج سمات الابعاد الخاصه بالارتفاع والعرض والطول الى تحديد قيمه وحدات البكسل

ويمكن لجميع السمات الاخرى ان تحتوي على قيمه yes او no ولكن اذا ظهرت السمه في الوسيط

يتم تعيين yes كقيمه افتراضيه في الوسيط ولذلك فان القيمه لا تكون بحاجه لتحديدها

مثال

}()function popup

،""،"window.open("popup.html

؛("top=40,left=40,width=200,height=100,status"

{

اغلاق الاطار المنبثق :=-

يمكن للاطار اغلاق نفسه من خلال التقنيه ()windoe.close

لكن يجب ان يتم تخصيص الاستدعاء الاول لتقنيه ()window.open لمتغير معين لانشاء كائن window

ويكتسب كائن window الجديد تقنيه ()windoe.close التي يمكن استخدامها في اغلاق اطار منبثق

مثال

المثال التالي يقوم بالتحقق من وجود اطار منبثق قبل اغلاقه عندما تخرج للمستخدم من الاطار الرئيسي

؛var popwindow

}()function popup

،"",'popwindow = window.open("popup.html

،top=40,left=40,width=200,height=100,directories"

؛("location,menubar,resizable,scrollbars,toolbar

{

} ()function close_popup

؛()if(popwindow != null)popwindow.close

{

؛window.onload = popup

؛window.onunload = close_popup

ضبط الحجم :=-

يمكن للمستخدم التحكم بدرجه وضوح الشاشه من خلال خصائص الارتفاع والعرض الخاصه بكائن

window.screen

مثال

في هذا المثال تقوم جافا سكربت بتحديد درجه الوضوح والحجم الداخلي للاطار لكل من المتصفحين

النت سكيب والانترنت اكسبلور

if(document.all){ //Internet Explorer

؛var win_w = window.document.body.clientWidth

؛var win_h = window.document.body.clientHeight

{

if(document.layers){ //Netscape

؛win_w = window.innerWidth

؛win_h = window.innerHeight

{

}(if(document.all || document.layers

؛var scr_w = window.screen.width

؛var scr_h = window.screen.height

؛var res = "Resolution:"+scr_w+"x"+scr_h

؛var iwin = "Innerwindow:"+win_w+"x"+win_h

؛(alert(res+"\n"+iwin

{

****
ولطباعه المستند في اطار المتصفح قم باستدعاء تقنيه window.print()
****

مصفوفه المقاطع :=-

يحتوي كائن window.frames على مصفوفه خاصه بجميع المقاطع في الاطارات التي تم تعريفها

داخل علامات ترميز frameset الموجوده في html

ويمكن للغه جافا سكربت ان تقوم بعرض علامات الترميز من داخل اي مقطع باستخدام تركيب

[]parent.window.frames

يمكن عرض المقاطع الفرديه باستخدام ارقام الفهارس الخاصه بها في

المصفوفه او القيمه التي تم تحديدها في سمه name داخل علامه الترميز frame التي

كتابتها بلغه html ولذلك فان اي مقطع يحمل اسم main يمكن ان يتم عرضه من داخل اي مقطع

اخر عن طريق parent.window.frames.main

مثال

":var framenums = "No.frames

؛parent.window.frames.length+

":var framenames = "\nFrame1

؛parent.window.frames[0].name+

":var framenames+= "\nFrame2

؛parent.window.frames[1].name+

":var framenames+= "\nFrame3

؛parent.window.frames[2].name+

؛(alert(framenums+framenames

كيفيه الوصول الى المقاطع داخل الاطار :=-

مثال

}()function next

؛"parent.frames.btm.window.location = "nextpage.htm

{

}()function write_header

؛"var msg = "Frame2
Written by Frame1

؛(parent.frames.hdr.window.document.write(msg

{

؛window.onload = write_header

معالجة الاخطاء :=-

يمكن تعيين داله مخصصه للتعامل مع الاخطاء لكائن window.onerror من اجل تصحيح الاخطاء

التي تحدث في وقت التشغيل

فعندما يواجه المتصفح ايه اخطاء فانه يقوم بالبحث عن معالج الاخطاء

في النص البرمجي ولكن اذا لم يجده فانه يقوم بمعالجه الاخطاء

بنفسه من خلال فتح مربع حوار browser error

وعند اكتشاف الخطا يتم عرض قيمه true في المتصفح للاشاره الى ان الخطا قد تم تصحيحه

في هذا المثال التالي يتم تعيين كل معلومه خاصه بالخطا في داله error-handler

من خلال المتصفح من اجل مصلحه المستخدم

```
window.onerror = errorhandler؛

}(function errorhandler(msg,url,ln

alert("Error:"+msg+"\nIn File:"+url+"\nAt Line:"+ln);  return true؛

{

This is a deliberate mistake//

؛"document.form[theMistake].value="myButton
```

-----------------*-----------------*-----------------------*-----------------

انتهى الدرس التاسع

وارجوا ان اكون قد افدتكم في هذا الدرس

ولا تنسونا من خالص دعائكم

ولا تنسونوا اخواننا المجاهدين في فلسطين وافغانستان وكشمير والشيشيان واندونيسا وفي كل مكان

من دعائكم في هذه الايام المباركه

تقبلوا تحياتي

اخوكم حمود العنزي

hma2003h911@yahoo.com